

**AMENDMENTS TO THE CLAIMS:**

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Currently Amended) A method for moving data objects in a computer system from a first storage location to a second storage location of a hardware memory device, the method comprising:

selecting a data object ~~from~~ stored in the first storage location, the data object being assigned to an identifier (ID);

~~assigning an identifier (ID) to the data object;~~

determining, using a processor, whether another process is attempting to perform a transaction with the data object by querying whether the ID is stored in a transactional type lock object;

upon determining that another process is not attempting to perform a transaction with the data object, storing the ID in [[a]] the transactional type lock object;

determining, using the processor, whether another process is moving the data object to a new storage location by querying whether the ID is stored in a permanent type lock object;

upon determining that another process is not moving the data object to a new storage location whether the ID is stored successfully in the transactional type lock object, and upon a successful storage, storing the ID in [[a]] the permanent type lock object, thereby indicating that the data object is stored at the first storage location;

determining whether the ID is stored successfully in the permanent type lock object, and upon a successful storage, deleting the ID from the transactional type lock object;

storing the data object at the second storage location;

assigning the ID to the data object stored in the second storage location ~~to the ID~~  
~~in the permanent type lock object;~~

deleting the data object from the first storage location; and

deleting the ID from the permanent type lock object, ~~thereby indicating that the data object is not stored at the first storage location,~~ after the data object has been deleted from the first storage location.

2. (Previously Presented) The method of claim 1, wherein the data object comprises one or more fields of one or more tables, and wherein the ID comprises one or more key fields of the one or more tables.

3. (Previously Presented) The method of claim 1, wherein the data object is stored in a file and wherein an assignment of the ID to the file or a name of the file is stored in the permanent type lock object.

4. (Canceled).

5. (Previously Presented) The method of claim 1, wherein storing the ID in the permanent type lock object comprises storing IDs of other data objects in the permanent type lock object before storing the data object at the second storage location.

6. (Canceled).

7. (Currently Amended) The method of claim 1, further comprising:  
checking whether the data object is ~~contained~~ stored in the second storage location and if the data object is ~~contained~~ stored in the second storage location, skipping storing the data object at the second storage location.

8. (Currently Amended) The method of claim 7, wherein ~~the checking whether the data object is stored in the second storage location~~ comprises querying whether the ID is stored in at least one of the transactional type lock object and the permanent type lock object.

9. (Currently Amended) The method of claim 1, further comprising:  
~~determining whether the ID was successfully stored in the transactional type lock object, and upon an unsuccessful storage,~~ checking whether the data object has been successfully stored in the second storage location, and if the data object has not been successfully stored in the second storage location, skipping deleting the data object

from the first storage location and skipping deleting the ID from the permanent type lock object.

10. (Currently Amended) The method of claim 1, for use in an enterprise resource planning software.

11. (Currently Amended) A computer system for processing data, the computer system comprising:  
memory means for storing program instructions;  
input means for entering the data;  
storage means for storing the data;  
a processor responsive to the program instructions, wherein the program instructions comprise program code means for performing a method for moving data objects in the computer system from a first storage location to a second storage location of the storage means, the method comprising:

selecting a data object ~~from~~ stored in the first storage location, the data object being assigned to an identifier (ID);

~~assigning an identifier (ID) to the data object;~~

determining whether another process is attempting to perform a transaction with the data object by querying whether the ID is stored in a transactional type lock object;

upon determining that another process is not attempting to perform a transaction with the data object, storing the ID in [[a]] the transactional type lock object;

determining whether another process is moving the data object to a new storage location by querying whether the ID is stored in a permanent type lock object;

upon determining that another process is not moving the data object to a new storage location ~~whether the ID is stored successfully in the transactional type lock object, and upon a successful storage, storing the ID in [[a]] the permanent type lock object, thereby indicating that the data object is stored at the first storage location;~~

determining whether the ID is stored successfully in the permanent type lock object, and upon a successful storage, deleting the ID from the transactional type lock object;

storing the data object at the second storage location;

assigning the ID to the data object stored in the second storage location ~~to the ID in the permanent type lock object;~~

deleting the data object from the first storage location; and

deleting the ID from the permanent type lock object, ~~thereby indicating that the data object is not stored at the first storage location,~~ after the data object has been deleted from the first storage location.

12. (Currently Amended) A computer readable storage medium comprising instructions for performing a method for moving data objects in a computer system from a first storage location to a second storage location of a hardware storage device, the method comprising:

selecting a data object from stored in the first storage location, the data object being assigned to an identifier (ID);

assigning an identifier (ID) to the data object;

determining, using a processor, whether another process is attempting to perform a transaction with the data object by querying whether the ID is stored in a transactional type lock object;

upon determining that another process is not attempting to perform a transaction with the data object, storing the ID in [[a]] the transactional type lock object;

determining, using the processor, whether another process is moving the data object to a new storage location by querying whether the ID is stored in a permanent type lock object;

upon determining that another process is not moving the data object to a new storage location whether the ID is stored successfully in the transactional type lock object, and upon a successful storage, storing the ID in [[a]] the permanent type lock object, thereby indicating that the data object is stored at the first storage location;

determining whether the ID is stored successfully in the permanent type lock object, and upon a successful storage, deleting the ID from the transactional type lock object;

storing the data object at the second storage location;

assigning the ID to the data object stored in the second storage location to the ID in the permanent type lock object;

deleting the data object from the first storage location; and

deleting the ID from the permanent type lock object, ~~thereby indicating that the data object is not stored at the first storage location~~, after the data object has been deleted from the first storage location.

13-14. (Canceled).

15. (Previously Presented) The computer readable storage medium of claim 12, wherein the data object comprises one or more fields of one or more tables, and wherein the ID comprises one or more key fields of the one or more tables.

16. (Previously Presented) The computer readable storage medium of claim 12, wherein the data object is stored in a file and wherein an assignment of the ID to the file or a name of the file is stored in the permanent type lock object.

17. (Canceled).

18. (Previously Presented) The computer readable storage medium of claim 12, wherein storing the ID in the permanent type lock object comprises storing IDs of other data objects in the permanent type lock object before storing the data object at the second storage location.

19. (Canceled).

20. (Currently Amended) The computer readable storage medium of claim 12, wherein the method further comprises:

checking whether the data object is ~~contained~~ stored in the second storage location and if the data object is ~~contained~~ stored in the second storage location, skipping storing the data object at the second storage location.

21. (Currently Amended) The computer readable storage medium of claim 20, wherein the checking whether the data object is stored in the second storage location comprises querying whether the ID is stored in at least one of the transactional type lock object and the permanent type lock object.

22. (Currently Amended) The computer readable storage medium of claim 12, wherein the method further comprises:

~~determining whether the ID was successfully stored in the transactional type lock object, and upon an unsuccessful storage,~~ checking whether the data object has been successfully stored in the second storage location, and if the data object has not been successfully stored in the second storage location, skipping deleting the data object from the first storage location and skipping deleting the ID from the permanent type lock object.



23. (Currently Amended) A computer system for processing data, the computer system comprising:

a processor executing program instructions;

means for selecting a data object ~~from stored in a~~ the first storage location of a hardware memory device, the data object being assigned to an identifier (ID);

~~means for assigning an identifier (ID) to the data object;~~

means for determining whether another process is attempting to perform a transaction with the data object by querying whether the ID is stored in a transactional type lock object;

means for storing the ID in [[a]] ~~the~~ transactional type lock object when it is determined that another process is not attempting to perform a transaction with the data object;

means for determining whether another process is moving the data object to a new storage location by querying whether the ID is stored in a permanent type lock object;

~~means for determining whether the ID is stored successfully in the transactional type lock object, and upon a successful storage, storing the ID in [[a]] the permanent type lock object, thereby indicating that the data object is stored at the first storage location~~ when it is determined that another process is not moving the data object to a new storage location;

means for determining whether the ID is stored successfully in the permanent type lock object, and upon a successful storage, deleting the ID from the transactional type lock object;

means for storing the data object at ~~the~~ a second storage location of the hardware memory device;

means for assigning the ID to the data object stored in the second storage location ~~to the ID in the permanent type lock object~~;

means for deleting the data object from the first storage location; and

means for deleting the ID from the permanent type lock object, ~~thereby indicating that the data object is not stored at the first storage location~~, after the data object has been deleted from the first storage location.

24. (Previously Presented) The computer system of claim 23, wherein the data object comprises one or more fields of one or more tables, and wherein the ID comprises one or more key fields of the one or more tables.

25. (Previously Presented) The computer system of claim 23, further comprising:

means for storing the data object in a file; and

means for storing an assignment of the ID to the file or a name of the file in the permanent type lock object.

26. (Canceled).

27. (Previously Presented) The computer system of claim 23, wherein the means for storing the ID in the permanent type lock object comprises means for storing IDs of other data objects in the permanent type lock object before storing the data object at the second storage location.

28. (Canceled).

29. (Currently Amended) The computer system of claim 23, further comprising:

means for checking whether the data object is ~~contained~~ stored in the second storage location and if the data object is ~~contained~~ stored in the second storage location, skipping storing the data object at the second storage location.

30. (Currently Amended) The computer system of claim 29, wherein the means for checking whether the data object is stored in the second storage location comprises means for querying whether the ID is stored in at least one of the transactional type lock object and the permanent type lock object.

31. (Currently Amended) The computer system of claim 23, further comprising:

means for ~~determining whether the ID was successfully stored in the transactional type lock object, and upon an unsuccessful storage,~~ checking whether the data object has been successfully stored in the second storage location, and if the data object has not been successfully stored in the second storage location, skipping deleting the data object from the first storage location and skipping deleting the ID from the permanent type lock object.

32-39. (Canceled).